

Course: Architecture and Programming Principles

Instructor: Professor Shaikh

Student Name: Yaoting Jiang

Date: Dec 1st, 2021

PAYMENT METHOD RESOURCES

Use Case: Get payment method

- **Resource(s):** payment method
- **Request URL:** http://localhost:5000/payment_method
- **Request body:** N/A
- **Response:**

The screenshot shows a REST client interface with the following details:

- Request:** GET http://localhost:5000/payment_method
- Status:** 200 OK, 27 ms, 344 B
- Response Body (JSON):**

```
{  "id": "61a7df80b3d272ee25081797",  "credit_card": "2000 0000 0000 0000",  "expiration_date": "11/11/21",  "security_code": "000",  "billing_address": "111 toyama dr, Sunnyvale, CA, 94089"}
```

Use Case: Create a payment method

- **Resource(s):** payment method
- **Request URL:** `http://localhost:5000/payment_method?credit_card=4400 8888 9999 0000&expiration_date=11/11/21&security_code=111&billing_address=160 Pleasant Dr, Boston, MA, 02148`
- **Request body:** N/A
- **Response:**

The screenshot shows a REST client interface with a POST request configured. The URL is `http://localhost:5000/payment_method?credit_card=4400 8888 9999 0000&expiration_date=11/11/21&security_code=111&billing_address=160 Pleasant Dr, Boston, MA, 02148`. The request body is empty. The response is a 200 OK status with a JSON body.

Request Details:

- Method: POST
- URL: `http://localhost:5000/payment_method?credit_card=4400 8888 9999 0000&expiration_date=11/11/21&security_code=111&billing_address=160 Pleasant Dr, Boston, MA, 02148`
- Params: credit_card (4400 8888 9999 0000), expiration_date (11/11/21), security_code (111), billing_address (160 Pleasant Dr, Boston, MA, 02148)

Response Details:

- Status: 200 OK
- Time: 8 ms
- Size: 341 B
- Save Response: [button]

Response Body (JSON):

```
1 {
2   "_id": {
3     "$oid": "61a7e0b88275062967d65f6a"
4   },
5   "credit_card": "4400 8888 9999 0000",
6   "expiration_date": "11/11/21",
7   "security_code": "111",
8   "billing_address": "160 Pleasant Dr, Boston, MA, 02148"
9 }
```

Use Case: Update a payment method

- **Resource(s):** payment method
- **Request URL:** `http://localhost:5000/payment_method/61a7df80b3d272ee25081797?credit_card=3300 0000 1111 2222&expiration_date=10/11/21&security_code=222&billing_address=222 toyama dr, Sunnyvale, CA 94089`
- **Request body:** shown in the screenshot below
- **Response:**

The screenshot displays a REST client interface with a PATCH request configured. The URL is `http://localhost:5000/payment_method/61a7df80b3d272ee25081797?credit_card=3300 0000 1111 2222&expiration_date=10/11/21&security_code=222&billing_address=222 toyama dr, Sunnyvale, CA 94089`. The request body is a JSON object with the following fields:

```
{  "_id": {    "$oid": "61a7df80b3d272ee25081797"  },  "credit_card": "1100 0000 3333 0000",  "expiration_date": "10/10/25",  "security_code": "000",  "billing_address": "360 Huntington Ave, Boston, MA, 02148"}
```

The response is a 200 OK status with a response time of 61 ms and a body size of 344 B. The response body is a JSON object with the following fields:

```
{  "_id": {    "$oid": "61a7df80b3d272ee25081797"  },  "credit_card": "1100 0000 3333 0000",  "expiration_date": "10/10/25",  "security_code": "000",  "billing_address": "360 Huntington Ave, Boston, MA, 02148"}
```

Use Case: Delete a payment method

- **Resource(s):** payment method
- **Request URL:**http://localhost:5000/payment_method/61a7df80b3d272ee25081797
- **Request body:** N/A
- **Response:**

GET http://localhost:50...

DEL http://localhost:50...

+ ...

No Environment

http://localhost:5000/payment_method/61a7df80b3d272ee25081797

Save

DELETE

http://localhost:5000/payment_method/61a7df80b3d272ee25081797

Send

Params

Auth

Headers (8)

Body

Pre-req.

Tests

Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body

Cookies

Headers (4)

Test Results

200 OK

18 ms

146 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

1

RATING RESOURCES

Use Case: Get rating

- **Resource(s):** rating
- **Request URL:** http://localhost:5000/rating
- **Request body:** N/A
- **Response:**

GET http://localhost:50... GET http://localhost:50... + ...

No Environment

http://localhost:5000/rating

Save

GET

http://localhost:5000/rating

Send

Params

Auth

Headers (8)

Body

Pre-req.

Tests

Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body

Cookies

Headers (4)

Test Results

200 OK

13 ms

236 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   {
3     "_id": {
4       "$oid": "61a971ce3a5e8c45587d4b55"
5     },
6     "score": 5,
7     "dog_walker": "TEST DOGWALKER"
8   }
9 }
```

Use Case: create a rating

- **Resource(s):** rating
- **Request URL:** `http://localhost:5000/rating?score=4&dog_walker=Test Dogwalker2`
- **Request body:** N/A
- **Response:**

The screenshot displays a REST client interface with a POST request to `http://localhost:5000/rating?score=4&dog_walker=Test Dogwalker2`. The request is configured with the method `POST` and the environment set to `No Environment`. The `Params` tab is active, showing query parameters `score=4` and `dog_walker=Test Dogwalker2`. The `Body` tab is also active, showing a JSON response in the `Pretty` view. The response status is `200 OK` with a response time of `14 ms` and a size of `235 B`. The JSON response is as follows:

```
{
  "_id": {
    "$oid": "61a974433a5e8c45587d4b56"
  },
  "score": 4,
  "dog_walker": "Test Dogwalker2"
}
```

Use Case: update a rating

- **Resource(s):** rating
- **Request URL:** `http://localhost:5000/rating/61a971ce3a5e8c45587d4b55`
- **Request body:** as shown in the screenshot below

Response:

The screenshot displays a REST client interface with a PATCH request configured for the URL `http://localhost:5000/rating/61a971ce3a5e8c45587d4b55`. The request body is a JSON object with the following structure:

```
1 {
2   "score": 3,
3   "dog_walker": "Test Dogwalker2"
4 }
```

The response is shown in the bottom panel, indicating a `200 OK` status with a response time of `26 ms` and a size of `235 B`. The response body is a JSON object:

```
1 {
2   "_id": {
3     "$oid": "61a971ce3a5e8c45587d4b55"
4   },
5   "score": 3,
6   "dog_walker": "Test Dogwalker2"
7 }
```

Use Case: delete a rating

- **Resource(s):** rating
- **Request URL:** http://localhost:5000/61a971ce3a5e8c45587d4b55
- **Request body:** as shown in the screenshot below

Response:

The screenshot displays a REST client interface with a DELETE request and its response. The top bar shows the request method as DELETE and the URL as http://localhost:5000/rating/61a971ce3a5e8c45587d4b55. The request body is set to JSON and contains the following content:

```
1 {
2   "score": 3,
3   "dog_walker": "Test Dogwalker2"
4 }
```

The response is shown in the bottom panel, indicating a 200 OK status with a response time of 66 ms and a body size of 146 B. The response body is currently empty, represented by an empty array [].

UPDATE DOG WALKER RESOURCES - ADDED PAGINATION AND FILTERING

Use Case: filter out dog walkers whose rating is 3

- **Resource(s):** dog walkers
- **Request URL:** `http://localhost:5000/dog_walker?rating=3`
- **Request body:** N/A

Response:

The screenshot shows a REST client interface with a GET request to `http://localhost:5000/dog_walker?rating=3`. The response is a JSON array of two dog walker objects, each with a rating of 3. The interface includes tabs for Params, Auth, Headers (8), Body, Pre-req, Tests, Settings, and Cookies. The Body tab is selected, showing the JSON response in a Pretty view.

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	rating	3			
	Key	Value	Description		

Body Cookies Headers (4) Test Results 200 OK 10 ms 590 B Save Response

Pretty Raw Preview Visualize JSON

```
16     "first_name": "test",
17     "last_name": "3",
18     "email": "dg3@test.com",
19     "phone": "8808889999",
20     "rating": 3
21   },
22   {
23     "_id": {
24       "$oid": "61aa79f278498392980020bd"
25     },
26     "first_name": "test",
27     "last_name": "4",
28     "email": "dg4@test.com",
29     "phone": "8808889933",
30     "rating": 3
31   }
32 }
```

Use Case: Get the first two dog walkers

- **Resource(s):** dog walkers
- **Request URL:** `http://localhost:5000/dog_walker?count=2`
- **Request body:** N/A

Response:

The screenshot shows a REST client interface with a GET request to `http://localhost:5000/dog_walker?count=2`. The response is a JSON array of two dog walker objects. The status is 200 OK, and the response time is 13 ms.

Request: GET `http://localhost:5000/dog_walker?count=2`

Query Params:

KEY	VALUE	DESCRIPTION
count	2	

Response Body (JSON):

```
3   {
4     "_id": {
5       "$oid": "61a6daeb015c79e63648f146"
6     },
7     "first_name": "test",
8     "last_name": "1",
9     "email": "dg1@test.com",
10    "phone": "6178889999",
11    "rating": 0
12  },
13  {
14    "_id": {
15      "$oid": "61aa74db77c686b727a2e457"
16    },
17    "first_name": "test",
18    "last_name": "2",
19    "email": "dg2@test.com",
20    "phone": "7168889999",
21    "rating": 3
22  }
```

Use Case: Get three dog walkers starting from 1

- **Resource(s):** dog walkers
- **Request URL:** `http://localhost:5000/dog_walker?from=1&count=3`
- **Request body:** N/A

Response:

The screenshot shows a REST client interface with a GET request to `http://localhost:5000/dog_walker?from=1&count=3`. The request is configured with the method GET and the URL. The response is a 200 OK status with a 10 ms response time and 590 B of data. The response body is displayed in JSON format, showing three dog walker objects.

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	from	1			
<input checked="" type="checkbox"/>	count	3			

Body

```
3  {
4    "_id": {
5      "$oid": "61aa74db77c686b727a2e457"
6    },
7    "first_name": "test",
8    "last_name": "2",
9    "email": "dg2@test.com",
10   "phone": "7168889999",
11   "rating": 3
12 },
13 {
14   "_id": {
15     "$oid": "61aa79d778498392980020bc"
16   },
17   "first_name": "test",
18   "last_name": "3",
19   "email": "dg3@test.com",
20   "phone": "8808889999",
21   "rating": 3
22 },
23 {
24   "_id": {
25     "$oid": "61aa79f278498392980020bd"
26   },
27   "first_name": "test",
28   "last_name": "4",
```

UPDATE DOG OWNER RESOURCES - ADDED PAGINATION AND SORTING

Use Case: Sort the dog owners by their last name

- **Resource(s):** dog owner
- **Request URL:** `http://localhost:5000/dog_owner?lastname`
- **Request body:** N/A

Response:

The screenshot shows a REST client interface with a GET request to `http://localhost:5000/dog_owner?lastname`. The response is a JSON array of two dog owner objects, sorted by last name. The status is 200 OK, with a response time of 13 ms and a size of 1.71 KB.

Request:

```
GET http://localhost:5000/dog_owner?lastname
```

Response Body:

```
[
  {
    "_id": {
      "$oid": "61aa877d96f11f1e84ecfe66"
    },
    "first_name": "Cody",
    "last_name": "Brian",
    "email": "cb@test.com",
    "phone": "3807770010",
    "street_address": "21 toyama dr",
    "city": "Sunnyvale",
    "state": "CA",
    "zipcode": "94089",
    "country": "USA",
    "credit_card_num": "4422 0000 0000 2211",
    "expiration": "03/12/22",
    "security_code": "122",
    "dog_name": "cindy",
    "dog_size": "big",
    "dog_sex": "female"
  },
  {
    "_id": {
      "$oid": "61aa877d96f11f1e84ecfe65"
    },
    "first_name": "Alice",
    "last_name": "Cohen",
    "email": "ac@test.com",
    "phone": "4007770000"
  }
]
```

GET http://localhost:50...

GET http://localhost:50...

+

...

No Environment

⌵

👁

http://localhost:5000/dog_owner?lastname

Save ⌵

🔗

💬

</>

GET ⌵ http://localhost:5000/dog_owner?lastname

Send ⌵

Params ● Authorization Headers (8) Body ● Pre-request Script Tests Settings Cookies

☑ lastname

Key	Value	Description
-----	-------	-------------

Body Cookies Headers (4) Test Results

🌐 200 OK 13 ms 1.71 KB Save Response ⌵

Pretty Raw Preview Visualize JSON ⌵

📄 🔍

```
24      "$oid": "61aa877d96f11f1e84ecfe65"
25    },
26    "first_name": "Alice",
27    "last_name": "Cohen",
28    "email": "ac@test.com",
29    "phone": "4807770000",
30    "street_address": "331 toyama dr",
31    "city": "Sunnyvale",
32    "state": "CA",
33    "zipcode": "94089",
34    "country": "USA",
35    "credit_card_num": "4400 0000 0000 1111",
36    "expiration": "11/12/22",
37    "security_code": "111",
38    "dog_name": "hans",
39    "dog_size": "big",
40    "dog_sex": "male"
41  },
42  {
43    "_id": {
44      "$oid": "61aa877d96f11f1e84ecfe64"
45    },
46    "first_name": "Jean",
47    "last_name": "Wang",
48    "email": "jw@test.com",
49    "phone": "4807773333",
50    "street_address": "222 toyama dr",
51    "city": "Sunnyvale",
52    "state": "CA",
53    "zipcode": "94089",
54    "country": "USA",
55    "credit_card_num": "4400 0000 0000 1111",
56    "expiration": "11/12/22",
57    "security_code": "111",
58    "dog_name": "hans",
59    "dog_size": "big",
60    "dog_sex": "male"
61  }
62 }
```

Use Case: Get the first dog owners

- **Resource(s):** dog owner
- **Request URL:** `http://localhost:5000/dog_owner?count=1`
- **Request body:** N/A

Response:

The screenshot displays a REST client interface with a GET request to `http://localhost:5000/dog_owner?count=1`. The response is a 200 OK status with a 14 ms response time and 545 B of data. The response body is shown in JSON format, detailing a dog owner's information.

Request: `GET http://localhost:5000/dog_owner?count=1`

Response: 200 OK 14 ms 545 B

Response Body (JSON):

```
{
  "_id": {
    "$oid": "61aa877d96f11f1e84ecfe66"
  },
  "first_name": "Cody",
  "last_name": "Brian",
  "email": "cb@test.com",
  "phone": "3807770010",
  "street_address": "21 toyama dr",
  "city": "Sunnyvale",
  "state": "CA",
  "zipcode": "94089",
  "country": "USA",
  "credit_card_num": "4422 0000 0000 2211",
  "expiration": "03/12/22",
  "security_code": "122",
  "dog_name": "cindy",
  "dog_size": "big",
  "dog_sex": "female"
}
```

I'm taking a grace period for this assignment

49-787 Request for Grace Period - Pass 3

yaotingj@andrew.cmu.edu [Switch account](#)

 Draft saved

Your email will be recorded when you submit this form

* Required

Use this form to redeem a pass for a 48-hour extension on a valid assignment per course syllabus. (Note: Requests for invalid assignments will not be honored, though your pass will be marked as used!)

Title of assignment for which grace period is requested *

Project 7B

Reason for requesting grace period (Do not put personal/private information) *

My ~~mongodb~~ is not working again for some re

A copy of your responses will be emailed to yaotingj@andrew.cmu.edu.

Submit

[Clear form](#)

Never submit passwords through Google Forms.

This form was created inside of Carnegie Mellon University Silicon Valley campus. [Report Abuse](#)